



MEMORIA DE DATOS.

La memoria de datos tiene una organización de 256x8. En esta memoria se guardan todos los datos que usamos para las variables y arreglos en un programa. La arquitectura de la memoria de datos se muestra en la figura 1.

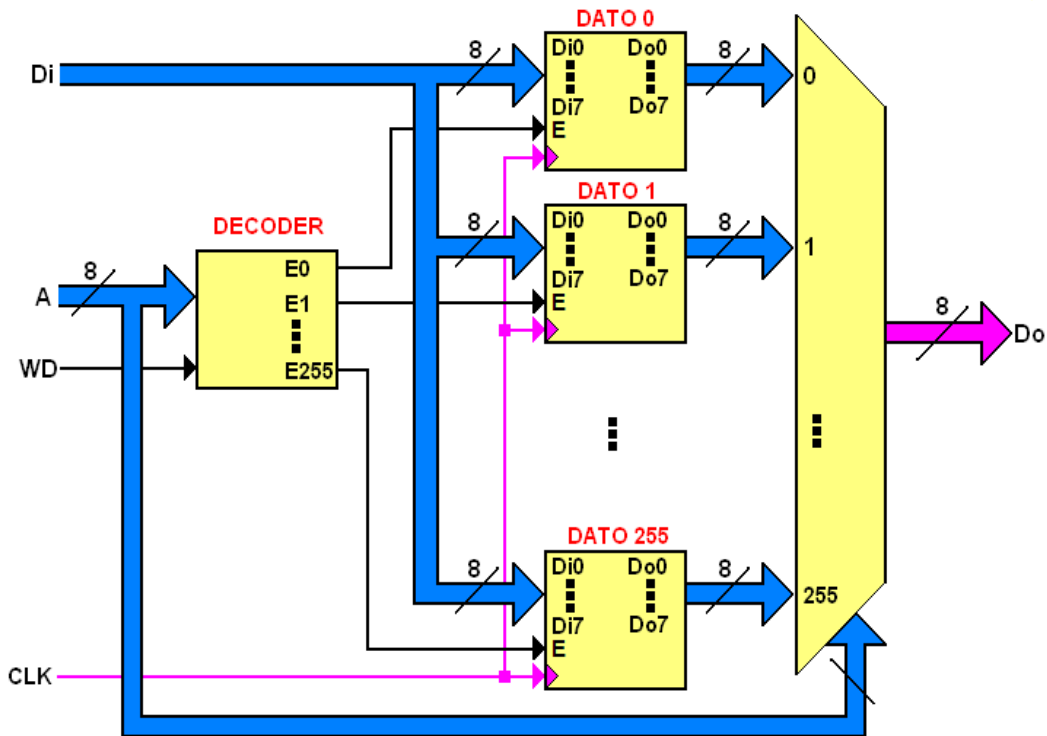


Figura 1: Memoria de Datos

Como se puede observar en la figura 1 la memoria de datos maneja las siguientes señales:

- **Bus de datos** (Do7,...,Do0). Este es un bus de salida de 8 bits. Este bus se utiliza para leer el dato contenido en la dirección indicada por el bus de direcciones.
- **Bus de datos** (Di7,...,Di0). Este es un bus de entrada de 8 bits. Este bus se utiliza para escribir un dato en la dirección indicada por el bus de direcciones.
- **Bus de direcciones** (A7,...,A0). Este es un bus de entrada de 8 bits con el cual indicamos la localidad de la memoria de datos que se va a leer o escribir. Con este bus podemos direccionar 256 localidades del mapa de la memoria de datos.
- **WD**. Señal de escritura para la memoria de datos (Write Data). Cuando esta señal tiene 1, se



realiza la escritura del dato colocado en el bus de datos de entrada (Di_7, \dots, Di_0) en la localidad de memoria especificada por el bus de direcciones (A_7, \dots, A_0). **Esta señal se ejecuta de forma síncrona, es decir, se ejecuta en el momento que llega una flanco de subida de la señal de reloj.**

➤ **CLK.** Señal de reloj. Las tareas del procesador se ejecutan en flanco de subida.

La lectura de la memoria se realiza de forma asíncrona siempre, por lo que no depende de la señal de reloj ni de la señal de control **WD**.

Los datos de la memoria $dato_0, dato_1, \dots, dato_{255}$ se encuentran implementados en una memoria RAM de un solo puerto: A. En el caso de los FPGA's de Xilinx los LUT's de los slices puede configurarse como una memoria Ram Distribuida de un solo puerto, por lo que se puede usar ese recurso dedicado para implementar la memoria de datos. La escritura de los datos se hace manera síncrona y la lectura se hace de manera asíncrona.